CLAIMS

What is claimed is:

1. A method for early packet steering by examining a packet as it is received at a routing location, thereby avoiding having to store the entire packet in memory prior to processing, said method comprising the steps of:

(1) parsing the packet as it is received to thereby extract portions of a packet header;

(2) processing the packet header in parallel before the arrival of a packet payload being carried by the packet; and

(3) steering the packet to a correct address without storing the entire packet payload in memory.

2. The method as defined in claim 1 wherein the method further comprises the step of collapsing a plurality of parsing decisions for the packet into a single operation;

3. The method as defined in claim 2 wherein the method further comprises the step of simultaneously performing a plurality of compare and branch operations in a single clock cycle to thereby parse a destination address of the packet.

4. The method as defined in claim 3 wherein the method further comprises the step of passing the packet to an output port immediately upon arrival of the packet payload.

5. The method as defined in claim 4 wherein the method further comprises the step of parsing the packet after arrival of the initial 64 bits of the packet header of the packet.

6. The method as defined in claim 5 wherein the method further comprises the step of determining before the first 64 bits have arrived if there is a problem with the packet which will preclude further processing of the packet.

7. The method as defined in claim 6 wherein the method further comprises the steps of:

(1) receiving the packet into an elastic buffer to thereby enable the system to perform its analysis before the entire packet arrives; and

(2) parsing information from the packet header in parallel, including destination address, source address, information to perform a cyclical redundancy check (CRC) and a hash table comparison, protocol analysis, and length analysis.

8 The method as defined in claim 7 wherein the method further comprises the step of compiling early packet steering statistics that are utilized to determine if the packet is valid, and a destination address for routing the packet.

9. The method as defined in claim 8 wherein the method further comprises the steps of:

(1) treating the destination address as an index that points to a physical port; and

(2) mapping directly to the physical port address utilizing content addressable memory techniques such that rather than billions of possible outcomes, a comparison is valid for a relatively small and manageable number of physical ports.

10. The method as defined in claim 9 wherein the method further comprises the step of utilizing the source address analysis to determine if an incoming packet has special requirements including special routing, special actions or rejection.

11 The method as defined in claim 10 wherein the method further comprises the step of performing the CRC calculation in hardware as the packet arrives, thereby removing that step from a host CPU and saving overhead, wherein the CRC can be performed as soon as the destination address has arrived in the packet.

12. The method as defined in claim 11 wherein the method further comprises the steps of:

(1) generating a status word, wherein the status word is utilized to transfer the packet to a correct destination address; and

(2) appending the status word to the packet so that it can be retrieved from as the packet as is necessary.

13. The method as defined in claim 12 wherein the method further comprises the steps of:

(1) storing a destination address in the status word;

(2) storing a source address in the status word;

(3) storing an alignment byte in the status word;

(4) storing a protocol field in the status word; and

(5) storing an end of packet field in the status word.

14. The method as defined in claim 13 wherein the method further comprises the step of reducing latency of the packet as it is received, thereby reducing a cost of the system because less memory is required for storing incoming packets, and less support circuitry is required to support the memory.

15. The method as defined in claim 14 wherein the method further comprises the steps of:

(1) reducing a total time between input and output of the packet; and

(2) reducing overhead on a system bus by parsing the packet header without having to send the packet header to the host CPU.